AJSLP

Research Article

Typical Toddlers' Participation in "Just-in-Time" Programming of Vocabulary for Visual Scene Display Augmentative and Alternative Communication Apps on Mobile Technology: A Descriptive Study

Christine Holyfield,[a] Kathryn Drager,[a] Janice Light,[a] and Jessica Gosnell Caron[a]

Purpose: Augmentative and alternative communication (AAC) promotes communicative participation and language development for young children with complex communication needs. However, the motor, linguistic, and cognitive demands of many AAC technologies restrict young children's operational use of and influence over these technologies. The purpose of the current study is to better understand young children's participation in programming vocabulary "just in time" on an AAC application with minimized demands.
Method: A descriptive study was implemented to highlight the participation of 10 typically developing toddlers (*M* age: 16 months, range: 10–22 months) in just-in-time vocabulary programming in an AAC app with visual scene displays.
Results: All 10 toddlers participated in some capacity in adding new visual scene displays and vocabulary to the app just in time. Differences in participation across steps were observed, suggesting variation in the developmental demands of controls involved in vocabulary programming.
Conclusions: Results from the current study provide clinical insights toward involving young children in AAC programming just in time and steps that may allow for more independent participation or require more scaffolding. Technology designed to minimize motor, cognitive, and linguistic demands may allow children to participate in programming devices at a younger age.

Young children with complex communication needs (CCN; i.e., children under the age of 5 whose speech does not meet their daily needs or whose speech development is at risk) benefit from augmentative and alternative communication (AAC) intervention (Romski, Sevcik, Barton-Hulsey, & Whitmore, 2015). AAC intervention promotes participation in communication for young children with CCN by resulting in an increase in communication turns, functions, and initiations (e.g., Dicarlo & Banajee, 2000) and interactions between parents and children (Light, Binger, & Kelford Smith, 1994). AAC also benefits language development for young children with CCN, including vocabulary development (Wright, Kaiser, Reikowsky, & Roberts, 2013) and grammatical development (Binger & Light, 2007). It is important to note there is no evidence that AAC intervention hinders speech development; in fact, there is evidence that AAC may modestly support speech development (Millar, Light, & Schlosser, 2006). In addition, AAC may have positive impacts on development beyond speech, language, and communication, such as motor development (Whitmore, Romski, & Sevcik, 2014). AAC can also replace challenging behavior as a means of communicating discomfort, frustration, or protest (Mirenda, 1997).

Limitations of AAC for Young Children

Despite the robust benefits of AAC for young children with CCN, current technologies may place unnecessary limitations on AAC users who are young children and/or at the beginning stages of communication development (Light & Drager, 2007). In particular, traditional AAC layouts and representations may not be ideal for young children. Grid layouts include rows and columns of content that can be selected. Representations commonly

[a]Pennsylvania State University, State College
Correspondence to Christine Holyfield: christineholyfield@gmail.com
Editor: Shelley Gray
Associate Editor: Cynthia Press
Received December 27, 2015
Revision received June 11, 2016
Accepted July 11, 2016
https://doi.org/10.1044/2017_AJSLP-15-0197

Disclosure: The authors have declared that no competing interests existed at the time of publication.

used in grid layouts include icons or symbols representing isolated objects or people or parts of objects or people (e.g., a hand reaching for a square to represent *want;* Worah, McNaughton, Light, & Benedek-Wood, 2015). Grid layouts require users to visually scan through content to find desired information. The representations often included in grid layouts require users to recognize concepts depicted outside of their natural (or any) context. Drager, Light, Curran Speltz, Fallon, and Jeffries (2003) found that typically developing 2- and 3-year-olds had minimal success locating vocabulary on a grid display with popular symbol representations and subsequently concluded that this technology was too demanding and therefore inappropriate for young children with CCN. It is unfortunate that, despite this conclusion, a recent survey revealed that grids are the most commonly utilized language organization technique when creating displays for young children and that representation is often considered less of a priority than other display features (Thistle & Wilkinson, 2015).

Potential Features of AAC Apps to Support Language Learning for Young Children

Visual Scene Displays

There are some newer developments in AAC technology that could support access to language for young children with CCN by minimizing demands. One such technology is visual scene displays (VSDs). This language organization option contrasts the traditional and frequently utilized grid display option. VSDs depict scenes of natural life events involving people (Light et al., 2004). The concepts, therefore, are grounded in the context in which they typically occur in young children's lives. Drager et al. (2003) argue that VSDs are inherently more context-rich than grids because "although concepts represented in a grid layout are separated in physical space, the symbols in an integrated scene [VSD] derive their meaning not only from their specific representations (e.g., a glass) but also from their relation within the rest of the scene (e.g., a drink on a table surrounded by breakfast foods, indicating orange juice)" (p. 300).

From a visual standpoint, locating concepts in VSDs may be less demanding for young children because they are more similar to the real world in which young children recognize, learn, and use concepts. From a cognitive and linguistic standpoint, the contextual nature of VSDs may facilitate the comprehension and use of AAC concepts for young children with CCN (Light & Drager, 2007). This may be particularly important for individuals in the earliest stages of language development for whom communication is contextually bound. Therefore, the built-in context of VSDs may also have positive implications for language development.

Figure 1 is an example of a children's book illustration that could be photographed and used as a VSD. The illustration depicts a young child peering out a window with wonder at a bird perched on a branch. Vocabulary could be added so that when selecting the child, the hotspot output is "Wow!" Vocabulary could also be added so that when selecting the bird, the VSD output is "Bird! Chirp, chirp!"

Figure 1. Example children's book illustration that could be photographed and captured as a visual scene display.

Although VSDs seem to be appropriate layout options for young children with CCN, programming VSDs within current AAC apps/technologies can be complex and time-consuming (Caron, Light, & Drager, 2016). The inundating nature of AAC programming creates a cascade of problems. Because of the large amount of time and effort required to program new content, professionals and other partners (e.g., parents) mostly program devices separate from the young children who use the device and separate from any interaction in which the young children use the content programmed. As a consequence, the adults programming content for young children are taxed with the impossible task of predicting all the vocabulary young children will need in any subsequent interaction and/or identifying all missed opportunities for communication in past breakdowns (Caron et al., 2016; Light, 1997)—not to mention the time requirements imposed onto an already busy group of care providers (Light & Kelford Smith, 1993).

Beyond—and perhaps more concerning than—the drawbacks of complex programming for partners, the paradigm of partners programming devices away from children and their interactions may also be problematic for young children who require AAC. In typical development, language is often learned by young children demonstrating interest in something in their natural environment, getting verbal feedback on the basis of the demonstrated interest, using newly acquired concepts, and getting feedback once again. For example, a child might point to a dog in the park. A parent would likely respond to the point with a statement such as "Dog!" The child might then repeat the parent (i.e., speak the word *dog*). The parent would then respond with another statement (e.g., "Right! That's a brown dog!"). Through the interaction, the child is able to build an association between the word and the referent. This sort of in-the-moment interaction with more expert partners is when learning often occurs (Rogoff, 1990). It is troubling to note that it is this sort of in-the-moment interaction to which young children with CCN who require aided AAC often do not have access (Light, Drager, & Currall, 2012). If the child or parent is not carrying a device containing the word *dog,* it would be a long process to add the word into the device. By the time the parent added the word, the child would likely have shifted interest to another event happening at the park. The large number of steps and large amount of time required to program many VSD apps and devices make it very difficult for partners to respond to the child's interest and allow him or her to communicate those aspects of naturally occurring contexts that happen to be of interest at any given moment.

Just-in-Time Programming

The concept of just-in-time programming refers to the programming of concepts that happen just as those concepts are needed by the individual to express—just in time (Schlosser et al., 2016). This conceptualization calls for technology that supports the creation of VSDs and other AAC content within interactions. We are thankful there is technology that currently exists that supports just-in-time programming. Caron et al. (2016) evaluated professionals' use of just-in-time programming when interacting with children using multiple AAC apps featuring VSD layouts. The professionals were able to program just in time across the different app conditions but were most effective in doing so using the app that was developed to minimize the steps and complexity of the programming process to support just-in-time programming, EasyVSD (under development by Invotek, http://www.invotek.org). EasyVSD is a communication app under development that recruits use of mobile technology's (i.e., a tablet's) onboard camera to quickly capture a photo that is used as a VSD. Then, with minimal steps, the VSD can be programmed with content. It is important to note that another study in which researchers used EasyVSD to program VSDs and related content for young children with CCN just in time found that the young children stayed highly engaged during times of just-in-time programming: 97% engagement during the addition of VSDs and 95% engagement during the addition of new vocabulary (Light et al., 2012).

AAC apps/technologies that allow for just-in-time programming do so by minimizing some of the programming demands originally placed by traditional AAC devices. However, it is not enough for communication partners to be able to quickly and easily program AAC content. With technological and design advancements, programming concepts within AAC devices/apps should become so simple and intuitive that young children and very early communicators can participate effectively in this process (Light, 1997). Although access to language is critical to communication, access to the programming of said language is also worthy of careful consideration (Light & Drager, 2007).

The Current Study

Despite the potential benefits of AAC apps/technology that support just-in-time programming of VSDs and vocabulary, to date no research has specifically investigated young children's participation in this process. There is evidence that professionals can effectively program just in time during interactions (Caron et al., 2016), but there is no current evidence that just-in-time programming can support young children's participation in selecting and adding concepts. Further, although features of AAC applications can be theoretically and developmentally driven to support just-in-time programming, no studies have evaluated the developmental demands of different programming features that may support just-in-time programming—a critical area of future research given the potentially limiting effects of demanding selection and programming (Drager et al., 2003; Light & Drager, 2007).

The current study was designed to investigate the viability of young children's participation in just-in-time programming and, to be specific, to explore the supports and constraints for typically developing young children (age 10–24 months). As there is no research in this area, an important first step is to gain a thorough understanding of the task demands from a developmental standpoint. A useful way to gain this understanding is to explore the ways and extent to which typically developing young children interact with just-in-time supportive technology. Typically developing toddlers between the ages of 10 and 24 months are in the early stages of linguistic development that VSDs are designed to support. Toddlers within this age range are on the cusp of developing first words or have developed first words and will soon begin to use two-word combinations. Their linguistic developmental status makes typical toddlers in this age range ideal for exploring programming using VSDs. They also are at an appropriate age to begin to understand the demands of vocabulary selection and programming for people in beginning developmental stages. Establishing these benchmarks with typical individuals can provide insight to the developmental appropriateness of apps that can be applied when moving forward with children with CCN (Higginbotham & Bedrosian, 1995). The current study does so by asking the following research question: In what percentage of opportunities do typically developing toddlers participate successfully within just-in-time programming steps required to create an AAC display containing a VSD and vocabulary concept?

Method

Participants

Ethics approval was obtained from the authors' university institutional review board prior to beginning the study. Participants were recruited through an online database of families interested or potentially interested in research and via flyers distributed at day care centers. Families identified through the database were contacted first by mail, then by phone or email. Signed consent was obtained from a parent or guardian prior to the start of the study. A total of 10 typically developing toddlers participated in the study (*M* age: 16 months, range: 10–22 months). Participating toddlers met the following criteria: They were (a) between 10 and 24 months of age at the start of the study, (b) reported by parents to have typical language and motor development and milestones, (c) reported by parents to have hearing and vision within normal limits, and (d) willing to interact with a computerized tablet.

After consent was obtained and before the start of the initial session, the parent or guardian present completed a short form inquiring about any developmental concerns that they had or that they were alerted to by a professional, such as a pediatrician. The form addressed vision, hearing, language, and motor development domains and milestones. No toddlers were disqualified from participating on the basis of the screening. Table 1 outlines some characteristics of the toddlers. Each participating toddler participated regularly in shared reading with parents on the basis of parent report. Parents for all of the toddlers reported mobile technology in the household and that the toddlers have had some interaction with the technology.

Materials

Books

The study took place within a shared book reading context. Book reading was an ideal context because of its developmental- and age-appropriateness with 10- to 24-month-olds. Also, storybooks provided pictures of interesting and familiar scenes that could be easily used as VSDs. In addition, they provided a consistent context for all sessions across participants. The toddlers chose familiar books from their homes or their day cares as the focus of sessions, and several popular board books were also provided by the researchers as additional options. As the majority of books utilized in the current study were books that were preferred by the toddlers at their homes or day cares, the books chosen by the toddlers were often different from the books chosen by other participating toddlers.

Tablet With VSDs and Just-in-Time Technology

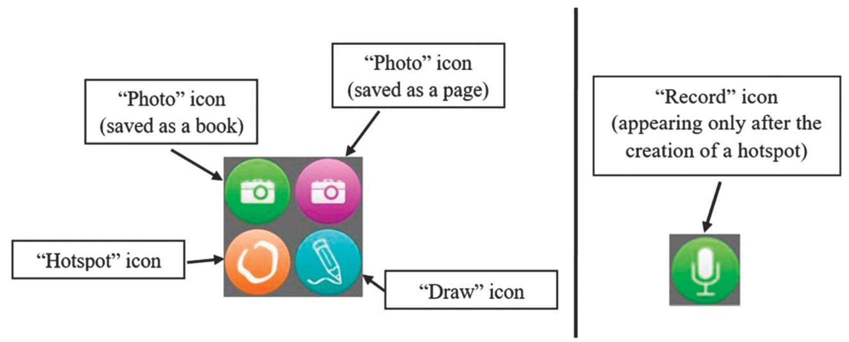
The toddlers interacted with a tablet containing an AAC app that allowed for the programming of VSDs and for communication using the VSDs. VSDs were appropriate representations for the toddlers because of their relative transparency (Drager et al., 2003). Version 1.37 of the EasyVSD software was utilized in the current study. The software featured just-in-time capabilities that were the focus of the current study. The app featured a menu on the left side that was consistent (i.e., it was always present with the available controls); this menu layout contrasts the traditional interface in which the menu is its own page that can be navigated to—sometimes with a watered-down version that is consistent on the screen. The buttons on the menu were also static; they did not change location. The buttons were also large in diameter to allow for easy activation, and the controls were represented with large icons that were easily recognizable or familiar due to popular app representations (e.g., the traditional camera icon to represent the button to take a photo). There were also a limited number of steps in creating a VSD and adding vocabulary. An onboard camera allowed photos to be captured and immediately made into VSDs. The minimum steps allowed programming vocabulary to happen quickly as well. Together, these simple features allowed EasyVSD to effectively support just-in-time programming. The EasyVSD software was run on an Android platform on a 10.1-in. Samsung tablet. See Figure 2 for a screenshot of the icons menu that appeared on the top left corner of the app screen (with icon labels added).

EasyVSD allowed users to create VSDs, program vocabulary to the VSDs, and draw on the VSDs. Adding a VSD, programming-related vocabulary concepts, and drawing involved the following steps: (1) selecting the photo icon, (2) touching the screen to take a photo, (3) selecting the hotspot icon, (4) drawing a circle around the desired area of the VSD to create a hotspot, (5) selecting the record icon, (6) immediately saying the message for the hotspot,

Table 1. Presence or absence of toddlers' demonstration of motor, linguistic, and cognitive behavioral markers during observation.

Participant	Age in months	Isolated point with movement and/or pressure	Coordinated gaze and movement	Triadic gaze	Following one-step directions	Use of symbolic communication	Use of gestural requests	Use of gestural comments	Use of spoken words
1	10	Yes	Yes	Yes	Yes	Yes	Yes	No	No
2	11	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
3	12	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
4	13	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
5	14	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
6	18	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
7	19	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
8	20	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
9	21	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
10	22	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Figure 2. The left pane depicts the static menu appearing in the top left corner of the app screen; the right pane depicts the record button, appearing after the creation of a hotspot (EasyVSD, Invotek).



(7) selecting the draw icon, and (8) drawing on the VSD. Adding a new VSD involved two steps (Steps 1 and 2); programming the new vocabulary involved four steps (Steps 3 and 4 to designate the hotspot and Steps 5 and 6 to record the word or phrase); and drawing involved two steps (Steps 7 and 8). Adding a VSD, programming related vocabulary, and drawing on the VSD required two types of steps: control functions and programming functions. Steps 1 (selecting photo icon), 3 (selecting the hotspot icon), 5 (selecting the record icon), and 7 (selecting the draw icon) all constituted control functions. Completing these steps did not complete a programming function; rather, these steps prepared the tablet for completing a programming function. The remaining steps—Steps 2 (touching the screen to take a photo), 4 (drawing a circle around the desired area of the VSD to create a hotspot), 6 (immediately saying the message for the hotspot), and 8 (drawing on the VSD)—were programming functions because they each completed some aspect of programming. Table 2 outlines the steps involved in the programming sequence, the physical behavior required to complete each of these steps, and some of the demands that were involved in completing each of the steps.

To create a VSD, the user first selected the photo icon (see Figure 2). After selecting the photo icon, the user would point the tablet at whatever scene he or she wanted to capture in the photo and touch the screen to take the photo (e.g., the user could take a photo of an illustration from a storybook of a birthday party). That photo then became the VSD. Next, to program vocabulary to the VSD, the user selected the hotspot icon (see Figure 2) then used his or her finger to outline the portion of the VSD that represents the vocabulary to be programmed (e.g., a little boy blowing out candles); a complete circle was not necessary for the hotspot to be created. The user would then select the record icon (which appears after a hotspot has been created; see Figure 2) and speak the message he or she wished the hotspot to yield (e.g., "Make a wish!"). Last, a user could select the draw icon (see Figure 2) and use his or her finger to draw on the hotspot (e.g., draw a number seven if it's the boy's seventh birthday). After this programming is complete, the app would display the VSD of the birthday party with the drawing of a number seven, and the user could touch the portion of the screen depicting the boy to hear the message, "Make a wish!"

Procedure

Each toddler participated in three sessions. The sessions lasted approximately 10 to 15 min. The procedures were the same across all three sessions. Across the sessions, the toddlers had between nine and 11 opportunities to participate in programming a new VSD and vocabulary concept; the toddlers had three to four opportunities for programming a scene and concept per session. This limited number per session was chosen to accommodate the toddlers' limited attention spans to a particular activity. The variation in three to four opportunities reflected the point in the session at which the toddler no longer indicated interest in participating (e.g., crawled off their mother's lap and began playing with a different toy). Although there was some redirection by the parents and the researcher throughout the three to four opportunities (e.g., "Wow! Look! It took a picture!"), no toddler disengaged from the activity to end the session before engaging in at least three opportunities. The interest drawn by the presence of the technology and the engaging interaction created by the researcher minimized disengagement for the time required to program at least three pages per session. Sessions occurred in a place that was familiar to the toddler and convenient for the family. For eight of the participating toddlers, the sessions occurred in their homes. For the other two toddlers, sessions occurred in a library area at their respective day cares. The setting did not change for any of the participants between sessions. Every session consisted of the toddler, a parent, and the second author. On occasion, a second parent or sibling was present as well, observing from a distance.

The toddlers sat on the lap of the parent present or, if preferred, sat next to the parent. The second author sat across from the toddler and parent and implemented all procedures. Prior to the start of the session, the researcher informed or reminded the parent that she was there to test

Table 2. Steps in programming in EasyVSD, required behaviors for completing the steps, and category of programming step.

Programming step	Behavior required	Examples of demands involved
Select photo icon	Touch photo icon	Use of an isolated point, activating screen within 0.75-in. icon diameter, cognitive understanding of the icon's representation
Take a photo	Touch anywhere on screen	Use of whole hand, part of hand, or an isolated point to activate the screen within the screen boundaries (10.1-in. width); use of appropriate timing to take the photo when the screen is aligned with the storybook illustration
Select hotspot icon	Touch hotspot icon	Use of an isolated point, activating screen within 0.75-in. icon diameter, cognitive understanding of the icon's representation
Make a hotspot	Draw any formed shape around target area, completion of shape into a full circle not required	Identification of appropriate portion of photo, use of an isolated point, creation of a circular or otherwise open shape to outline appropriate portion of the photo (although full enclosure is not required)
Select record icon	Touch record icon	Use of an isolated point, activating screen within 0.75-in. icon diameter, cognitive understanding of the icon's representation
Record hotspot message	Speak message within 2 s of selecting record icon	Use of speech and expressive language to speak message, use of appropriate timing to speak message immediately after the record icon is selected
Select draw icon	Touch draw icon	Use of an isolated point, activating screen within 0.75-in. icon diameter, cognitive understanding of the icon's representation
Draw	Touch anywhere on visual scene display	Use of whole hand, part of hand, or an isolated point to activate the screen within the visual scene display boundaries (width of the screen, minus the menu bar)

the software, not to test the child, and so there was no need to try to help by telling the toddler what to do. These instructions at the beginning of each session were sufficient in fielding any prompts from the parents beyond general redirections (e.g., "Look! We can pick a new book!"). Also, during the few sessions when a sibling was present in the room, the parents were very conscious of keeping the child busy with other activities while the toddler participated in the session (e.g., "Go back to coloring. Your brother is busy."). Throughout the sessions, parents served as a familiarity and a comfort and occasionally to redirect interest to the tablet and/or storybook, but they did not participate in the interaction. The researcher and the toddler interacted surrounding the storybooks and the programming.

Sessions began with the parent and researcher providing the toddler with options for storybooks to look at and read. The book chosen by the toddler served as the context for programming until the end of the session, until preference for another book was indicated by the toddler, or if the toddler was redirected back to the activity by being shown a different book (e.g., "Oh! This book looks funny! We could take a picture of this one!"). After a book was chosen by the participant, the investigator sat across from the toddler who was sitting on his or her parent's lap, and the book sat between them. The researcher held the tablet upright close to her face at midline with the toddler to make it accessible to the toddler from a motor and vision standpoint and to minimize joint attention demands (Light et al., 2012).

The researcher then engaged the toddler by reading the story. If there was a page that the toddler turned back to, spent a long time looking at, or had a reaction to (e.g., laughed at a picture), the researcher then asked the toddler, "Should we take a picture of that one?" If the toddler indicated that he or she did not want to take a photo of that page of the book, the researcher continued reading the story with the toddler. If the toddler did indicate that he or she wanted to take a photo of that book page (e.g., by looking up at the researcher, by smiling, by nodding), the researcher would use that page as the context for the VSD to be programmed for the model.

The investigator initiated the model by indicating to the toddler that she was going to demonstrate use of the tablet (e.g., "We're going to use the computer to talk about the books. Watch me!"). She then modeled each step in the sequence of programming a new VSD, adding a related vocabulary concept, and drawing on the VSD using the book chosen by the toddler (see Table 2 for an outline of each programming step and the programming behaviors required to complete each step). The researcher talked through each step as she completed it. The language she used during the talk-through of the model matched the language used to prompt the toddlers throughout each session. For example, when the researcher completed the "select photo icon" step during the model, she would say, "I'm going to tell the computer to take a picture." No explicit goal of the programming was relayed to the toddlers before or after the model because (a) the toddlers were all very interested in the story reading, the technology, and the programming features itself, so they did not need this sort of goal-oriented motivation, and (b) the metacognitive and metalinguistic skills for talking about goal orientation were likely beyond the current developmental profile of many of the toddlers.

After modeling the programming sequence, the investigator alerted the toddler that it was his or her turn to try, and they could find another page to take a picture. For this first and the subsequent two to three programming sequences, the investigator began by following the same procedures as the model for following the child's lead to determine an interesting and motivating book page to use as the context for each VSD.

Once the toddler chose a book page, the investigator began by setting up the opportunity for the toddler to initiate the first step in the programming sequence (e.g., "OK, let's take a picture of that one. Tell the computer you want to take a picture."). After setting up the opportunity for the toddler to complete the step, the investigator waited for 4 s or until the toddler completed or attempted to complete the step. After the 4 s, the investigator provided a verbal and gestural prompt to the toddler (e.g., saying "Tell the computer!" while pointing to the photo icon). Again, the investigator waited 4 s. After 4 s, the investigator modeled the step for the toddler. The investigator then waited an additional 4 s for the toddler to complete the step. If the toddler did not complete the step after the 4 s, the investigator completed the step and set up the opportunity for the next step. If at any point the toddler completed the step, the investigator provided positive feedback (e.g., "You did it!"). If at any point the toddler attempted the step, but did not complete it successfully, the investigator provided positive feedback (e.g., "Oh! You almost got it!") and completed the programming step. These procedures were repeated for each step in the programming sequence. Procedural reliability for the initial model; provision of opportunities, prompting, modeling wait time, and feedback for each VSD; and the provision of choice relative to the creation of each new VSD and each scene captured was 83% (range: 77%–92%) for a randomly selected 30% of opportunities.

Language for setting up the initial opportunity for each programming step was conceptual rather than behavioral in nature. For example, the initial opportunity for each of the control function steps followed the same structure: "We want to ____________. You can tell the computer." (take a picture, make a hotspot, make it talk, draw). Then, the programming functions were all simple directives (e.g., "You can take a picture" or "You can draw"). Although the language was conceptual and may have been difficult for the youngest toddlers to understand, an attempt was made to simplify the language and grammatical structures and shorten the length of the input provided by the investigator.

Data Measures and Analysis

Data were recorded online by the first author who was present at every session. The sessions were also videotaped for reliability and procedural fidelity purposes and so that the first author could revisit any missing or uncertain information. Each session, the video camera was positioned behind and slightly beside and above the toddler to capture the toddlers' physical interaction with the tablet (i.e., the video angled down above the toddler to view the researcher, the tablet screen in full as well as any selections made by the toddler).

Measures Coded

Each toddler's participation was coded for each step in the programming sequence (eight steps) for each of the nine to 11 opportunities provided. For each step, toddler performance was coded as successful participation, unsuccessful participation, or no response. Rate of successful participation relative to the number of opportunities offered was variable of interest that is reported in the results. Frequencies and percentages of successful turns were summarized for each step for each toddler to determine the viability of programming participation with young children within the developmental period of 10–22 months.

Successful participation. Steps were coded as successful participation if at any point in the prompting hierarchy (i.e., after the initial opportunity, after verbal and gestural prompting, after a model) the toddler completed the step in a way that allowed the researcher and the toddler to move on to the next step (i.e., the step did not need to be corrected or reattempted)—that is, the toddler made an onscreen activation that resulted in the adequate completion of the target step. For the control functions, this required the toddler to activate the appropriate button on the screen. For the "taking a photo" step, this required the toddler to activate anywhere on the screen. For the "making a hotspot" step, this required the toddler to draw a circle or other amorphous shape around an area of the hotspot that was large enough to be usable for communication by the toddler. For the "record" step, this required the toddler to speak a message that was usable for hotspot content (e.g., vocalizations were not counted). For the "draw" step, this required the toddler to drag a finger anywhere on the VSD. Within the draw step, toddlers also had the option of selecting different colors from a menu.

No response. Steps were coded as no response if the toddler did not interact with the tablet following the 4-s wait time after the model—that is, if the toddler did not activate the touch screen at any point.

Unsuccessful participation. Participation was coded as unsuccessful participation if the toddler activated the screen in some way but did not successfully complete the step. Steps that were unsuccessful included turns in which toddlers activated the incorrect button, drew hotspots that were too small to be usable, or took photos that were blurry or were not focused on the book.

Reliability of the Data

Reliability was completed by a master's student for one randomly selected session per toddler. The student coded the first three sequences of these sessions, accounting for 30% of the data. The first author provided training on coding for the master's student. Before completing reliability on the randomly selected sessions, the student participated in calibration on one of the videos not selected for reliability. Coding disputes during calibration were discussed until consensus was reached. The student then completed reliability on the randomly selected 30% of all programming opportunities. Inter-rater reliability was 95% (range: 92%–100%).

Results

Overall Participation

The toddlers participated successfully, on average, in 41% of opportunities (out of an average of 80 opportunities, range: 72–88). Of the steps in which the toddlers participated successfully, they completed 71% of those steps with prompting or modeling from the clinician. Despite overall frequent rates of successful participation, there was considerable variation in rates of successful participation (from 16% to 73%). Overall rates of successful participation for each of the 10 toddlers in ascending age order were 20%, 16%, 19%, 19%, 31%, 68%, 29%, 60%, 70%, and 73%. None of the toddlers participated successfully in the completion of each step continuously to create, program a VSD with content, and draw on it. Table 3 outlines rates of successful participation for each of the toddlers in each of the programming steps. The table also aggregates the average rate of successful participation both for each toddler and each step in the programming sequence.

Participation Across Programming Steps

The toddlers' successful participation varied considerably across the steps in the programming sequence (range: 3%–82%). The variation across steps appears greater than the variation across toddlers. It is clear that some steps were more difficult for the young children than others (e.g., only 3% overall participation for recording a message). Opposite that, the children successfully participated in some steps at high rates (e.g., drawing on the VSD). There were two steps in which the toddlers participated with an average of 75% accuracy or higher: taking a photo and drawing on the VSD. These are also the two steps in which all 10 toddlers successfully participated. There were four steps with which the toddlers participated with less than 75% but greater than 25% average accuracy: selecting the photo icon, selecting the hotspot icon, creating a hotspot, and selecting the record icon. For each of these steps, eight, six, seven, and six toddlers experienced successful participation, respectively. There were two steps in which the toddlers averaged less than 25% successful participation: selecting the record icon and recording. Five toddlers successfully selected the record icon; three toddlers successfully recorded a message.

Discussion

Overall Participation

Each toddler successfully participated in some capacity in programming VSDs just in time—even toddlers as young as 10 months of age. This finding is an important one because it suggests that, given access to simplified technology designed with developmental demands in mind, individuals at developmentally very early stages can participate in the programming process with support (i.e., wait time, prompting, and modeling from the researcher). Strategies such as those implemented by the researcher would be important when teaching any group to program an unfamiliar app.

On the basis of the results, it seems that the age of the toddlers may have had some influence over their rates of successful participation. In general terms, average rates of successful participation in programming steps for the older five toddlers (68%, 29%, 60%, 70%, and 73%) appear higher than the average rates of successful participation demonstrated by the younger five toddlers (20%, 16%, 19%, 19%, and 31%). Although the app was designed to minimize developmental demands in programming, it was not possible to eliminate motor, cognitive, and linguistic demands completely from the programming toddlers. It is likely that the older toddlers who have had longer to develop in these areas were, as a result, more readily equipped to meet the demands. In addition to being more prepared for the demands of the app, the older toddlers may have faced less difficulty comprehending the talk-alouds, opportunities, and prompts spoken by the researcher. Attempts were made to minimize the language demands imparted by the researcher's speech, but just as the demands of the app could only be minimized, not eliminated, so too were the language demands placed by the researcher's speech.

Participation Across Programming Steps

In addition to the toddler's point in development, successful participation seems to have also been affected

Table 3. Percentage of successful participation for each toddler out of total opportunities for participation in each programming step.

Participant	Age in months	Selecting photo icon	Taking a photo	Selecting hotspot icon	Creating a hotspot	Selecting record icon	Recording message	Selecting draw icon	Drawing on photo	*M* (*SD*)
1	10	11	67	0	11	0	0	0	67	20 (30)
2	11	0	50	0	0	0	0	0	80	16 (31)
3	12	10	70	0	10	0	10	0	50	19 (26)
4	13	0	80	0	0	0	0	0	70	19 (35)
5	14	27	100	18	0	0	0	18	82	31 (39)
6	18	100	90	80	30	70	0	80	90	68 (35)
7	19	30	40	10	10	20	0	30	90	29 (28)
8	20	82	91	45	45	18	9	91	100	60 (35)
9	21	80	100	70	80	30	10	90	100	70 (32)
10	22	90	100	70	70	60	0	100	90	73 (33)
M (*SD*)		43 (40)	79 (22)	29 (33)	27 (29)	20 (26)	3 (5)	41 (44)	82 (16)	41 (28)

by the differing demands imparted by each of the programming steps. It seems the toddlers had more difficulty participating successfully during some programming steps compared with others. These differences were likely due to task differences among steps or the interplay between the tasks and the profiles of participating toddlers. It seems that taking a picture and drawing on the VSD were the easiest for the toddlers. It also seems that selecting the record icon and recording a message were the hardest steps for the toddlers. The remaining four steps fell in between those groups. Below, speculative explanations about differences observed in toddler performance across these groupings are considered.

Taking a Picture and Drawing on the VSD

Touching anywhere on the screen (10.1 in.) resulted in successful completion of the "take a photo" step; the camera captured a photo upon any part of the screen being activated. In a similar manner, touching anywhere on the VSD resulted in successful completion of the "draw" step (the width of the screen minus the menu bar); the VSD was marked with color wherever a person touched on it. Although all of the toddlers demonstrated an isolated point, one was not required for either of these steps. Taking a photo also required timing with the alignment of the camera to the desired screen as well as an activation that was not so weighted that it moved the tablet and blurred the photo. However, the researcher was holding the tablet throughout the sessions and was able to align the camera with the scene, minimizing much of the timing demands for the toddlers. So these steps placed minimal motor demands on the toddlers. The steps also placed minimal linguistic demands on the toddlers; they were not required to interpret a representation other than one from the immediate context (i.e., the scene to be captured in a photo). The steps, combined with researcher support, also placed minimal cognitive demands on the toddlers. The toddlers could quickly touch the screen and may not have understood the reason they were touching the screen if they completed the steps following a gestural prompt or model.

Selecting the Photo Icon, Selecting the Hotspot Icon, Making a Hotspot, and Selecting the Draw Icon

All of the steps involving icon selection placed the linguistic demand on the toddlers of having to interpret the representations differentiating each icon. This may have in part accounted for the seemingly lower rates of success the toddlers experienced with these steps compared with the steps outlined above. However, if the toddlers completed the icon steps after a gestural prompt or model, they may not have needed to interpret the representations. The icon steps and creating a hotspot also placed heightened motor demands on the toddlers compared with the above two steps. The toddlers were required to activate a much smaller area of the screen (i.e., 0.75-in. diameter) in the case of the icon steps or drag their finger around an area of the VSD when creating a hotspot. In a cognitive manner, the icon steps required the toddlers to indicate to the computer that they wanted to complete the next step (e.g., taking a photo, drawing). The "make a hotspot" step required the toddlers to identify the area of the photo they wished to highlight although the toddlers may not necessarily have understood the purpose of the hotspot when circling an area of the VSD. For any of the steps, the cognitive demands were partially alleviated if the toddler completed the steps following gestural prompting or modeling from the researcher. For example, the child didn't necessarily need to comprehend the camera representation of the photo icon if the researcher just pointed to the icon he or she then was able to activate.

Selecting the Record Icon and Recording a Message

The average percentage for selecting the record icon was slightly lower than for the other icon steps. This may be just a chance difference. It also may have been due to researcher procedural inconsistency; because activation of this icon led immediately into recording a message, it was more often completed by the researcher without following each step in the prompting hierarchy than the other icon steps. From a motor, linguistic, and cognitive standpoint, this step was equivalent to the above steps. However, the "record hotspot message" step was more demanding in every aspect. Speech requires more advanced motor movement than use of the hands to activate the screen (the motor movement required in the other steps). Only three toddlers successfully completed this step. This was a programming step for which procedural fidelity was lower than the other steps; the toddlers may have demonstrated more success given an opportunity with each step of the prompting hierarchy every time. Given that eight of the 10 toddlers in the study had spoken words, more may have demonstrated success given more opportunities with the support outlined in the procedures. Speech is also more linguistically demanding than the other steps. Speech is a cognitively demanding task as well—especially the coordination of this speech within 4 s of the selection of the record icon or a prompt or model from the researcher.

Implications for AAC for Young Children With CCN

Provision of Opportunities for Programming Participation

Recent research has indicated that professionals can successfully program VSDs just in time during interactions with children (Caron et al., 2016). Also, when just-in-time programming occurs during interactions with young children, they appear to stay engaged (Light et al., 2012). On the basis of results from the current study that each participating toddler experienced some success in participating in the programming process, it may be possible for partners (e.g., family members, speech-language pathologists) to further the inclusion of children with CCN in the just-in-time programming process. It may be possible for partners to include young children with CCN in the physical act of programming. The participants in the current study were typically developing, so future research is needed to determine the feasibility and supports required to include young children with CCN in just-in-time programming. However, the young age of the toddlers in the current study suggests that individuals who require AAC may benefit from opportunities to be involved in the programming process at much earlier stages of development than may have been previously thought. Clinicians could provide such opportunities accompanied with the appropriate supports for children who require AAC to determine on a client-by-client basis the opportunities and supports that might lead to success in programming participation for each individual.

Also, the current study involved just-in-time programming of VSDs because they are traditionally time-consuming to program and seem to be the most appropriate layout option for young children and other individuals at beginning stages of development (Light & Drager, 2007). However, the concept of making AAC technology work just in time is much broader than programming content into VSDs (Schlosser et al., 2016). Future research could evaluate the benefits of incorporating older individuals who use grid displays in just-in-time programming of content onto those layout options.

If future research and clinical experience reveals that young children (or older individuals) with CCN can be successfully involved in the programming of AAC apps/devices, there may be some real benefits for people who use AAC in this involvement for language development, operational competence, and self-determination. Some possible benefits are outlined below.

Language development. Within the typical paradigm of AAC programming, particularly for young children with CCN, vocabulary is mostly added away from children who use AAC (Caron et al., 2016; Light, 1997). This paradigm is a restrictive one as it functionally excludes young children who use AAC from the sort of interaction in which language development typically occurs (i.e., a child demonstrates interest in something in the environment, a more expert partner provides language input relative to that demonstrated interest, the child goes on to use the newly acquired concept expressively and gets input as to the accuracy of its use—all within naturally occurring contexts).

It is fortunate that just-in-time programming seems to be a step toward solving this problem; professionals are able to complete just-in-time programming in interactions with children (Caron et al., 2016), and young children with CCN seem to benefit from just-in-time programming (Light et al., 2012). On the basis of the large increase in vocabulary expressed by young children with CCN in the study completed by Light et al. (2012), it seems probable that just-in-time programming supports language development. However, given the results of the current study, future research should examine if there are additional benefits on language development when including individuals with CCN who are early language learners in just-in-time programming. Children who use AAC tend to be ascribed to more of a respondent role in interactions, including storybook interactions (Light & Kelford Smith, 1993). However, through participating in programming during storybook or other interactions, it may be possible to support children who require AAC into taking more of a dominant role. In this role, they may be able to direct the conversation on the basis of their interests. For instance, they could choose which pages of a book become a VSD on the app as the typical toddlers did in the current study. Given the importance of children's attention and motivation and partner responsivity to language learning, supporting them in a more active role may support language development as well. Future research should explore the impact of involving young children with CCN and older individuals with CCN who are at early stages of language learning on outcomes related to language development (e.g., vocabulary acquisition and use).

Development of operational competence. For individuals who use AAC, there are four aspects of communication competence: social competence (i.e., understanding and use of language in communicative contexts), linguistic competence (i.e., understanding and use of the linguistic code), strategic competence (i.e., understanding and use of strategies to achieve communicative success despite linguistic, operational, and social limitations), and operational competence (i.e., understanding and use of AAC systems and their features; Light, 1989). For young children with CCN, typical AAC device operational demands are cumbersome and preclude young children with CCN from participating in vocabulary selection and programming (Light & Drager, 2007). Many features exclude typical young children from successful operations as well (Drager et al., 2003; Light et al., 2004). Results from the current study suggest that those same features that support just-in-time programming for adult communication partners (Caron et al., 2016) can also support participation in vocabulary selection and programming for young children. Involvement early on in AAC operations may also support the development of operational competence later in life. This development of operational competence, in conjunction with the other aspects of communication competence and overall competence, is critical to achieving communication competence (Light, 1989). Future research should explore the benefits of involvement in just-in-time programming on the short- and long-term operational competence of early language learners who require AAC and how operational competence may scaffold users' experience of competence in the other facets (i.e., linguistic, strategic, social).

Self-determination. In the traditional paradigm for programming AAC systems, choices about vocabulary and representations occur away from young children with CCN, and decisions are often adult-focused, not child-focused (Caron et al., 2016; Light, 1997; Light et al., 2004). This paradigm excludes the young children from making decisions around the vocabulary available in their systems and limits their power to direct interactions. However, results from the current study provide initial evidence that it may be possible to support young children in having a more prominent role in this decision-making process. Provision of such opportunities during just-in-time programming may promote self-determination by giving young children practice exercising decision making around their own communication, a central aspect of their life. Future research should explore the benefits of involvement in just-in-time programming on evidence of self-determination in young children with CCN and older individuals who require AAC and are early language learners.

Considering Variation in Performance and Demands

Although the results of the current study suggest that individuals who require AAC might experience success in active involvement in just-in-time programming at young ages and early developmental stages, it is critical to consider variation both in individual performance and in the demands of different technology and different programming steps. There was a large amount of variation in the rates of successful participation across toddlers in the current study.

Such differences across steps may affect both the expectations and support provided by clinicians when including young children with CCN in just-in-time programming. For instance, a step suspected to be more cognitively demanding may require talk-alouds about its completion or other verbal scaffolds. A step that seems to be more difficult from a motor standpoint may require clinicians to provide a gestural model to support the individual in performing the motor behavior, or it may require accommodations or partner-assisted completion (i.e., the individual may touch the screen but miss the icon, so the clinician provides feedback and selects the icon to continue the programming process). For example, a young child with Down syndrome may be able to activate small buttons on a touch screen with scaffolding (e.g., pointing toward the button). Opposite that, a child with cerebral palsy who has limited fine motor control may not have the isolated point necessary to activate small buttons accurately. In this case, the child may benefit from a stylus to activate the screen during programming if his or her grip is more reliable than his or her point. Future research could determine what scaffolds might be most appropriate or effective for individuals with CCN who have different motor, linguistic, and cognitive characteristics.

Clinicians may find that not all steps are feasible for involving young children with CCN or that opportunities for the completion of some steps result in more frustration than success. It may benefit clinicians to consider the demands involved in the completion of steps and compare those demands with the motor, cognitive, and linguistic profile of the person who uses AAC. It may be sensible to provide frequent opportunities and more independent practice for young children with CCN to complete those steps that seem to be a good match between the task demands and the child's characteristics. It may also make sense for the partner to complete those steps in which the child does not experience successful programming despite appropriate support. For instance, if the child with cerebral palsy above does not experience success using the stylus to activate a button, steps requiring the activation of the button may be completed by the clinician and steps that can be completed using a whole hand may be ideal opportunities for involving the child. Last, it may be that even with the provision of appropriate support and identification of the least-demanding steps involved in programming a specific AAC app or device, the app or device is too demanding for a young child who uses AAC overall. For this reason, young children with CCN may benefit from clinicians considering potential for involvement in programming and other operations when selecting AAC technologies. Given variability noted in the current study, future research should continue to explore the demands imparted in AAC operations across a range of technology and operational steps. Future research could evaluate the performance of young children with CCN in completing a variety of programming behaviors across features theoretically designed to minimize demands. Future research could also provide insight into partner (speech-language pathologists, other professionals, family members, and peers) behaviors that are most facilitative of the success of young children with CCN in programming.

AAC Design

The purpose of AAC technology could be stated as providing individuals access to communication and participation in those interactions in which their speech restricts them from fully participating. However, these same considerations of access should be extended to ensure that operations on AAC technologies (e.g., programming) are accessible to the spectrum of profiles of individuals utilizing them, including young children with CCN (Light, 1997) and other individuals with CCN who are at early stages of development. Advances in technology may make this ideal more feasible. For instance, the app in the current study utilized larger buttons, which allowed the toddlers to experience at least some success with steps involving this control. However, future technology may be able to minimize the necessity of buttons by requiring another behavior with fewer motor demands to complete the step. The possibilities of making AAC programming more accessible for young children with CCN are vast. For instance, if an individual has difficulty activating a touch screen, AAC technology could pick up on users' gestures or movements that do not require contact with the screen. In addition, the toddlers in the current study very infrequently participated in recording a hotspot. This finding is unsurprising; the toddlers had limited speech and expressive language. Technology that recognizes photos or concepts within photos could be used to automatically assign realistic-sounding synthesized speech to hotspots identified by the toddlers rather than requiring the speech to be recorded.

Limitations and Future Directions

Although typical toddlers' interaction with the just-in-time technology and participation in just-in-time programming is a logical and useful first step in understanding the developmental demands of features that support this participation, the largest limitation of the current study was that it did not involve young children with CCN. Potential intrinsic (e.g., cognitive, motor, language) factors of young children with CCN could affect their participation in just-in-time programming. Future research could seek to understand the ways in which these factors in young children with CCN resulting from a variety of disabilities (e.g., autism spectrum disorder, Down syndrome) affect just-in-time programming, and any bottlenecks or areas of difficulty can be supported or bypassed. Future research could also compare and manipulate different feature options to establish strong information about evidence-based AAC system design with the goal of accessibility to both communication and device operations.

The small sample size was another limitation of the current study; only 10 toddlers participated. Due to the small sample size, the data did not allow for statistical analyses that may have provided more confidence to the reader than comparing overall means alone. However, some of the large differences observed (e.g., 3% vs. 82%) provide reasonable evidence for differences in performance. The toddlers spanned 14 months of difference in age, and those months are associated with different cognitive, linguistic, and motor profiles. The age difference likely accounted for the variation in performance found in the data. Therefore, the wide inclusion criteria relative to age is another limitation of the current study. However, it allowed for description of performance participating in just-in-time programming across individuals with different developmental profiles (see Table 1).

Another major limitation of the study was the limited amount of information gathered about the participants. Although parent report was gathered in a way that would include any concerns voiced to the parents by professionals or that they themselves had about the development of their toddler, there is much variability in the typical development of children. Therefore, richer descriptions of the language, motor, and cognitive profiles of the participants would have allowed for a more in-depth discussion about the characteristics of the participants and how those characteristics may have affected their participation with the app and provide initial guidance on supporting young children with CCN as they participate in programming on the basis of intrinsic characteristics. Future research evaluating the demands of AAC technology could gather more detailed information about the participants through a variety of mechanisms, including the administration of standardized tests.

Further, the somewhat low procedural fidelity (83%) is another limitation of the current study. It was probably lower than preferred due to the fluid and flexible nature of storybook interactions with toddlers and because the toddlers benefited from support in the programming process. Investigator support for the toddlers (i.e., verbally and visually prompting and modeling) was included in the procedures, but the prompting hierarchy was not always implemented with the minimum 4-s wait time prescribed. For this reason, data relative to difference in toddler performance across levels of prompting hierarchy were not reported; the toddlers may have performed at more independent levels given more opportunities for participation while withholding prompting or modeling for the prescribed amount of time. This is unfortunate as this information could have provided clinicians with a starting point for thinking about appropriate supports when providing young children with CCN opportunities to participate in programming. Also, toddlers were not always provided with independent opportunities to respond. The reported percentage of participation that occurred after prompting and modeling may have been lower if this opportunity were more carefully provided; this percentage should be interpreted with caution.

Last, the software utilized in the current study offered features that supported just-in-time programming. However, although this study suggests features of the software are more developmentally appropriate than many current options available, the features are not exhaustive, and they are not perfect. Future research could examine the developmental appropriateness of more device features or potential device features. Future research should explore these potential feature supports for young children with CCN as well as older beginning communicators and test their utility and viability in real-life contexts.

Conclusion

With access to developmentally appropriate technology and partner support, typically developing toddlers as young as 10 months of age were able to participate successfully in just-in-time programming of VSDs and embedded vocabulary. Given the successful participation of the young toddlers, it seems that technology designed to support just-in-time programming may also be facilitative of access to programming for young children or individuals in early developmental stages (e.g., motor, linguistic). If future research reveals that young children with CCN and older individuals with CCN who are in early stages of development are successful in participating in just-in-time programming, their participation may have important benefits, such as positive impacts on language development, operational competence, and self-determination. Future research should assess the effects of just-in-time programming involvement for young children with CCN on these and other outcomes. Future research should continue to work toward technology that is accessible to all people who require AAC, for both communication and operations such as programming representations and content.

Acknowledgments

This research was funded through National Institutes of Health Grant 1R43HD059231-01A1, SBIR Phase II:“PlayTalk” Communication Software for Children with Complex Communication Needs, National Institute of Child Health and Human Development. In addition, Christine Holyfield and Jessica Gosnell Caron were also supported by funding from the Penn State AAC

Leadership Project, a doctoral training grant funded by U.S. Department of Education Grant H325D110008.

References

Binger, C., & Light, J. (2007). The effect of aided AAC modeling on the expression of multi-symbol messages by preschoolers who use AAC. *Augmentative and Alternative Communication, 23,* 30–43.

Caron, J., Light, J., & Drager, D. (2016). Operational demands of AAC mobile technology applications on programming vocabulary and engagement during professional and child interactions. *Augmentative and Alternative Communication, 32,* 12–24.

Dicarlo, C., & Banajee, M. (2000). Using voice output devices to increase initiations of young children with disabilities. *Journal of Early Intervention, 23,* 191–199.

Drager, K., Light, J., Curran Speltz, J., Fallon, K., & Jeffries, L. (2003). The performance of typically developing 2 ½-year-olds on dynamic display AAC technologies with different system layouts and language organizations. *Journal of Speech, Language, and Hearing Research, 46,* 298–312.

Higginbotham, D., & Bedrosian, J. (1995). Subject selection in AAC research: Decision points. *Augmentative and Alternative Communication, 11,* 11–13.

Light, J. (1989). Toward a definition of communicative competence for individuals using augmentative and alternative communication systems. *Augmentative and Alternative Communication, 5,* 137–144.

Light, J. (1997). Let's go star fishing: Reflections on the contexts of language learning for children who use aided AAC. *Augmentative and Alternative Communication, 13,* 158–171.

Light, J., Binger, C., & Kelford Smith, A. (1994). Story reading interactions between preschoolers who use AAC and their mothers. *Augmentative and Alternative Communication, 10,* 255–268.

Light, J., & Drager, K. (2007). AAC technologies for young children with complex communication needs: State of the science and future research directions. *Augmentative and Alternative Communication, 23,* 204–216.

Light, J., Drager, K., & Currall, J. (2012, November). *Effects of AAC systems with "just in time" programming for children with complex communication needs.* Presentation at the annual convention of the American Speech-Language-Hearing Association, Atlanta, GA. Retrieved from http://aac.psu.edu/?p=1281

Light, J., Drager, K., McCarthy, J., Mellott, S., Millar, D., Parrish, C., . . .Welliver, M. (2004). Performance of typically developing four- and five-year-old children with AAC systems using different language organization techniques. *Augmentative and Alternative Communication, 20,* 63–88.

Light, J., & Kelford Smith, A. (1993). Home literacy experiences of preschoolers who use AAC systems and their nondisabled peers. *Augmentative and Alternative Communication, 9,* 10–25.

Millar, D., Light, J., & Schlosser, R. (2006). The impact of augmentative and alternative communication intervention on the speech production of individuals with developmental disabilities: A research review. *Journal of Speech, Language, and Hearing Research, 49,* 248–264.

Mirenda, P. (1997). Supporting individuals with challenging behavior through functional communication training and AAC: Research review. *Augmentative and Alternative Communication, 13,* 207–225.

Rogoff, B. (1990). *Apprenticeship in thinking: Cognitive development in social context.* Oxford, United Kingdom: Oxford University Press.

Romski, M., Sevcik, R., Barton-Hulsey, A., & Whitmore, A. (2015). Early intervention and AAC: What a difference 30 years makes. *Augmentative and Alternative Communication, 31,* 181–202.

Schlosser, R. W., Shane, H. C., Allen, A. A., Abramson, J., Laubscher, E., & Dimery, K. (2016). Just-in-time supports in augmentative and alternative communication. *Journal of Developmental and Physical Disabilities, 28,* 177–193.

Thistle, J., & Wilkinson, K. (2015). Building evidence-based practice in AAC display design for young children: Current practices and future directions. *Augmentative and Alternative Communication, 31,* 124–136.

Whitmore, A., Romski, M. A., & Sevcik, R. (2014). Early augmented language intervention for children with developmental delays: Potential secondary motor outcomes. *Augmentative and Alternative Communication, 30,* 200–212.

Worah, S., McNaughton, D., Light, J., & Benedek-Wood, E. (2015). A comparison of two approaches for representing AAC vocabulary for young children. *International Journal of Speech-Language Pathology, 17,* 460–469.

Wright, C., Kaiser, A., Reikowsky, D., & Roberts, M. (2013). Effects of a naturalistic sign intervention on expressive language of toddlers with Down syndrome. *Journal of Speech, Language, and Hearing Research, 56,* 994–1008.